

Wichtige Methoden der Klasse String

charAt(), length(), replace(), split(), trim()

Auszug API-Dokumentation (überarbeitet):

public char charAt(int index)

Returns the char value at the specified index. An index ranges from 0 to length() - 1. The first char value of the sequence is at index 0, the next at index 1, and so on, as for array indexing.

Specified by:

Parameters: index - the index of the char value.

Returns: the char value at the specified index. The first char value is at index 0.

Beispiel: "Hallo".charAt(1) liefert ein 'a' als Charakter zurück

public int length()

Returns the length of this string. The length is equal to the number of Unicode code units in the string.

Specified by: length in interface CharSequence

Returns: the length of the sequence of characters represented by this object.

Beispiel: "Hallo".length() liefert den Integer 5 zurück

public String replace(char oldChar, char newChar)

Returns a new string resulting from replacing all occurrences of oldChar in this string with newChar.

If the character oldChar does not occur in the character sequence represented by this String object, then a reference to this String object is returned. Otherwise, a new String object is created that represents a character sequence identical to the character sequence represented by this String object, except that every occurrence of oldChar is replaced by an occurrence of newChar.

Parameters: oldChar - the old character. newChar - the new character.
HINWEIS: GEHT AUCH MIT GANZEN STRINGS (=Charakter Arrays!)

Returns: a string derived from this string by replacing every occurrence of oldChar with newChar.

Beispiel: "Hallo".replace('a', 'x') liefert den String "Hxllø" zurück.
„Hallo“.replace(„ll“, „xx“) liefert den String „Haxxo“ zurück

public String [] split(String regex)

Splits this string around matches of the given regular expression.

Trailing empty strings are not included in the resulting array.

Parameters: regex - the delimiting regular expression

Returns: the array of strings computed by splitting this string around matches of the given regular expression

Beispiel: "HalloxWeltxwiexgehts?".split("x") liefert ein String-Array mit 4 Elementen zurück:
{„Hallo“, „Welt“, „wie“, „gehts?“}

public String trim()

Returns a copy of the string, with leading and trailing whitespace omitted.

This method may be used to trim whitespace (as defined above) from the beginning and end of a string.

Returns: A copy of this string with leading and trailing white space removed, or this string if it has no leading or trailing white space.

Beispiel: " Hallo Welt!".trim() gibt "Hallo Welt!" zurück – ohne führende und folgende Leerzeichen oder Zeilenumbrüche.